PB83-156778

Creating and Documenting Procedures
Letting George Do It

(U.S.) Environmental Monitoring Systems Lab.
Research Triangle Park, NC

1983

U.S. Department of Commerce
National Technical Information Service
NTIS

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completi

1. REPORT NO.	2.	3.
EPA-600/D-83-006	PUBLISHED PAPER	PB83-156778

4. TITLE AND SUBTITLE	5. REPORT DATE
Creating and Documenting Procedures -- Letting George Do It --	1983
	6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S)	8. PERFORMING ORGANIZATION REPORT NO.
Dr. Lynn H. Wright	
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT NO.
Pollutant Analysis Branch (MD-78) Environmental Monitoring Division Environmental Monitoring Systems Laboratory/RTP U. S. Environmental Protection Agency Research Triangle Park, North Carolina 27711	11. CONTRACT/GRANT NO.
12. SPONSORING AGENCY NAME AND ADDRESS	13. TYPE OF REPORT AND PERIOD COVERED
same as above	14. SPONSORING AGENCY CODE EPA/600/08

15. SUPPLEMENTARY NOTES

16. ABSTRACT

This paper describes a method of creating and documenting automated data processing routines for the Finnigan INCOS gas chromatograph/mass spectrometer data system. A computer program in the form of a set of macro-instructions for the data system text editor is used to automatically convert a file written in a documentation format into executable procedures. An example of use of the program in the generation of automated procedures is given. Listings of the text editor routines are available from the author.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161

18. DISTRIBUTION STATEMENT	19. SECURITY CLASS *(This Report)*	21. NO. OF PAGES
	Unclassified	7
	20. SECURITY CLASS *(This page)*	22. PRICE
	Unclassified	

EPA Form 2220-1 (9-73)

EPA-600/D-83-006
PUBLISHED PAPER

CREATING AND DOCUMENTING PROCEDURES
-LETTING GEORGE DO IT-

By Lynn H. Wright
EMSL (MD-78)
U. S. Environmental Protection Agency
Research Triangle Park, NC 27711

This paper has been reviewed by the Environmental Monitoring Systems Laboratory, U. S. Environmental Protection Agency, and approved for publication. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

The PROCEDURE capability of MSDS provides the user with a means of creating complex automated routines for acquiring and processing mass spectral data. The simplest route to the development of a procedure is keyboard entry of the desired command sequences. However, drawbacks to this approach appear when modifications to an existing procedure are necessary or when complex procedures, i.e. procedures which call other procedures, are being developed. The problem is compounded when changes in one part of a complex procedure necessitate changes in interrelated procedures. Other problem areas are protection and documentation. The inadvertant destruction or redefinition of another user's procedure is all too easy in a multiuser environment. The MSDS TRACE/H listing, while easy to obtain, is at best a debugging aid; it is much too garbled to satisfactorily document a complex procedure.

An ideal situation should feature simplicity in both the creation and modification of procedures. Documentation should be easy to produce, maintain, and decipher. And last, but not least, only a minor effort should be required to recover a damaged procedure.

George's Editor (GE), the IDOS text editor, can be used to modify or even create .PR files. This approach may facilitate the editing of simple procedures, but not complex ones, and the problems of documentation and vulnerability remain. To overcome these drawbacks the process can be taken one step further by using GE to create and modify routines, even highly complex ones, as a single text or .DS file. Turning a .DS file into the requisite .PR files is simple and fast if you let George (GE) do it.

A q-register instruction set, listed in Figure 1, has been devised to let George do it. This routine, entitled PRLOAD, requires a small amount of formatting to create a procedure.DS file, since each procedure* in the file must begin with the character sequence:

"<cr><24 spaces>[prname.PR "

and end with the sequence:

"[END prname.PR]<cr>".

Embedded spaces within the quotes above are significant. Since the text within square brackets becomes a comment in the .PR file, the first sequence must eventually be closed with a right bracket. PRLOAD, upon execution, converts these comments into file descriptors and writes each record bounded by these comments on the disc as a separate .PR file. PRLOAD and its loader routine can be stored at the end of each procedure.DS file for easy access. An example of formatting for PRLOAD is shown in Figure 2.

In addition to separate documentation, several other practices insure the viability of procedures and other user files. In a multiuser environment it is helpful to prefix user filenames with the originator's initials and to include the version number in the filename. For example, our in-house descriptor for the above version of PRLOAD is LWPRLOAD8.DS. Retaining the main procedure name as a part of all subprocedure names, as illustrated in Figure 2, expedites the identification of the components of complex routines. It also simplifies changing version numbers within a procedure.DS file. For example, the GE command "J<CFIND3$FIND4$>$" would update the version number for all of the procedures in the procedure.DS file of Figure 2.

Another significant document control practice is the inclusion of the program revision number and date as a comment both in the program source and in the header of any hardcopy generated by the program.

Using GE to create procedures in a .DS file has several advantages. First is the ease of creation and correction offered by the text editor. Second, using the .DS files as a source for all of the components of a procedure provides one all-encompassing document for complex routines which is automatically updated whenever the routine is updated (all updates and corrections should be performed on the procedure.DS file and PRLOAD used to generate the .PR files). As a bonus, the transfer of complex routines from one data system to another is simplified by this approach, since only one file, procedure.DS, need be copied.

* NOTE: MSDS requires a 24 character leader, which may or may not occupy 24 print spaces, at the beginning of each .PR or .ME file. This leader is not treated as part of the procedure command string.

```
[ TO LOAD POSITION CURSOR AT START OF THIS LINE & ENTER "3XXQX$" ]
.U23LS[ BEGIN PRLOAD ]$OL.UOS[ END PRLOAD ]$.U1QO,Q1XV$Q2,Q1K$
[ TO EXECUTE PRLOAD "EY" THE APPROPRIATE .DS FILE, THEN "QV$" ]

[       PRLOAD8 7/13/82]

[ BEGIN PRLOAD ].UO.ZUZ<QOJ:S
                        [ $;[ IF "<CR><24 SP><L BRAC><SP>" NOT FOUND END;
   ELSE
   ].U1:S.PR$;[ IF ".PR" CAN'T B FOUND END; ELSE
      ] U2Q2-Q1-10"L[ IF PRNAME.PR HAS A VALID # OF CHARS THEN
         ]OL.U3Q1,Q2XX[ STORE PRNAME.PR
         ]QZJI:S[ END $GX@I/ ]$U4$/$OLXYK$[ CREATE A SEARCH SUBR. IN Y
         ]QOJQYQ4;[ IF "<L BRAC> END PRNAME.PR <R BRAC>" NOT FOUND END;
         ELSE
            ]13,13I$.U5[ ADD 2 <CR>'S
            ]ZJIEO$GX@I/$Q3,Q5W$EC$/$[ CREATE SUBR TO WRITE PRNAME.PR
            ]I^A$GXI WRITTEN$13,27I$[ & FLASH MESSAGE
            ]-L2XW2KQW[ & EXECUTE SUBR
            ]Q5J-2D.UO'[CLEAN UP & SET POINTER TO END OF PRNAME.PR
      ]Q2-Q1-9"G[ IF NAME.PR IS NOT A VALID # OF CHARS LONG
         ]Q2UO'[ POSITION CURSOR WHERE "BEGIN" SEARCH ENDED & LOOP
]$>[ END PRLOAD ]
```

Figure 1. PRLOAD preceeded by its loader routine. (A listing with extensive comments may be obtained from the author upon request.)

```
                                [ FIND3.PR  LOCATE AND QUANTITATE PEAKS IN
                                GC RUN USING CHRO & REL RET TIMES ]
SET1 !10;                       [ SCAN # OF INT. STD.  PREVIOUSLY PUT IN 10 ]
GETL;                           [GET NEXT ENTRY FROM THE LIBRARY]
CHRO(I;R;#;%;$;N5,10;A-1,4<;O;G-15,15;D-30,30;E)    [QUANTITATE THE NEXT ]
ERASE                           [ PEAK(S); PUT RESULTS INTO THE TEMP .GL]
FIND3A;
LOOP;                           [REPEAT FOR ALL LIBRARY ENTRIES]
[ END FIND3.PR ]


                                [ FIND3A.PR  HOW MANY FOUND? ]FIND3B;
EDQL,$(-;N;#;A;E);              [ WE GET HERE FROM FIND3C IF NO PEAKS FOUND ]
[ END FIND3A.PR ]


                                [ FIND3B.PR ]FIND3C;
EDQL ,$(A;E);    [WE GET HERE FROM FIND3C IF 1 PEAK FOUND ]
RETURN FIND3A;
[ END FIND3B.PR ]


                                [ FIND3C.PR ]
[ TEST FOR HOW MANY PEAKS FIND3 QUANTITATED FOR ONE LIBRARY ENTRY. ]
IF FIND3B !28,  #1;
EDQL ,$ (U*100;A;E);      [ PUT LARGEST PEAK IN TEMP.GL INTO FINAL GL ]
RETURN FIND3A;
[ END FIND3C.PR ]
```

Figure 2. Example of a Complex Procedure in Text Form. PRLOAD would create files FIND3.PR, FIND3A.PR, FIND3B.PR, and FIND3C.PR from this text. (Example extracted from AUTOQU.PR)